

IN THE CLAIMS:

1. (Currently amended) A method of processing work items in a data processing system comprising the steps of:

generating an interrupt in response to receipt of a work item in the system;  
disabling system interrupts;  
~~servicing the generated interrupt to schedule~~ scheduling a task through the generated interrupt  
for later processing of the work ~~item; item;~~ without re-enabling the interrupt;  
subsequently executing the task to process the work item; and  
processing additional work items received by the system; and  
when there are no additional work items for processing, speculatively scheduling a further task for processing of any subsequently received work items ~~that are subsequently received~~ in the system, without enabling system interrupts.

2. (Currently amended) ~~A~~ The method as claimed in of claim 1, further comprising the ~~further~~ steps of:

executing the speculatively scheduled task to process ~~any~~ work items received by the system;  
enabling system interrupts when ~~on a determination that there are no~~ additional work items have been received by the system when the speculatively scheduled task is executed; ~~to be processed;~~  
~~enabling the interrupt;~~ and

processing one or more work items when at least one work item has been received by the system when the speculatively scheduled task is executed; and

~~on a determination that there are work items to process;~~ speculatively scheduling a an additional further task for processing of subsequently received work items after processing the one or more work items, without ~~re-enabling the interrupt;~~ enabling system interrupts.

3. (Currently amended) ~~A~~ The method as claimed in of claim 1, wherein the work items are managed on a queue.

4. (Currently amended) ~~A~~ The method as claimed in of claim 1 ~~where in~~ 1, wherein the event

that further work items are received after the task is scheduled and prior to execution of the task, the step of executing the task comprises processing all the received work items.

5. (Currently amended) A data processing system comprising:

processing means for executing tasks to process work items in the data processing system; and interrupt generating means for generating an interrupt in response to receipt of a work item in the system; wherein the processing means is operable to:

disable system interrupts;

~~service the generated interrupt to~~ schedule a task through the generated interrupt for later processing of the work ~~item; item, without re-enabling the interrupt;~~

~~subsequently~~ execute the task to process the work item; and

process additional work items received by the system; and

when there are no additional work items for processing, speculatively schedule a further task for processing of any subsequently received work items ~~that are subsequently received~~ in the system, without enabling system interrupts.

6. (Currently amended) ~~A~~ The data processing system ~~as claimed in~~ of claim 5, the processing means being operable ~~on a determination that there are work items to be processed to~~ execute the speculatively scheduled task to process the work items received by the system, and to ~~schedule a further speculative task; and operable on a determination that there are no work items to be processed to enable the interrupt.~~ enable system interrupts when no additional work items have been received by the system when the speculatively scheduled task is executed, process one or more work items when at least one work item has been received by the system when the speculatively scheduled task is executed, and speculatively schedule an additional further task for processing of subsequently received work items after processing the one or more work items, without enabling system interrupts.

7. (Currently amended) ~~A~~ The data processing system ~~as claimed in claim 5~~ of claim 5, further ~~including~~ comprising memory for storing the received work items on a queue.

8. (Currently amended) ~~A The data processing system as claimed in claim 5~~ wherein of claim 5, wherein the event that further work items are received after the task is scheduled and prior to execution of the task, the processing means is operable to execute the task to process all the work items.

9. (Currently amended) ~~A The data processing system as claimed in claim 5 of claim 5,~~ wherein the interrupt generating means and processing means are embodied in a data storage controller and the work items comprise data transfer requests from an attached host system.

10. (Currently amended) A computer program product comprising a computer usable medium having computer readable program code means embodied in the medium for processing work items in a data processing system, the program code means comprising:

code means for causing the data processing system to generate an interrupt in response to receipt of a work item in the system;

code means for causing the data processing system to disable system interrupts;

code means for causing the data processing system to ~~service a generated work item interrupt~~ to schedule a task through the generated interrupt for ~~later~~ processing of the work ~~item; item, without re-enabling the interrupt;~~

code means for causing the data processing system to ~~subsequently~~ execute the task to process the work item; ~~and~~

code means for causing the data processing system to process additional work items received by the system; and

code means for causing the data processing system to speculatively schedule a further task for processing of any subsequently received work items ~~that are subsequently received~~ in the system when there are no additional work items for processing, without enabling system interrupts.

11. (Currently amended) ~~A The computer program product as claimed in of claim 10,~~ the computer readable program code means further comprising:

code means for causing the data processing system to execute the speculatively scheduled

task to process ~~any~~ work items received by the system; and

code means for causing the data processing system to enable ~~the interrupt on a determination that there are~~ system interrupts when no additional work items for processing have been received by the system when the speculatively scheduled task is executed;

code means for causing the data processing system to process one or more work items when at least one work items has been received by the system when the speculatively scheduled task is executed; and

code means for causing the data processing system to speculatively schedule an additional further task for processing of subsequently received work items after processing the one or more work items, without enabling system interrupts.

12. (Previously presented) A method of processing work items in a data processing system, comprising:

effectively providing an interrupt-based mechanism for processing work items, when system utilization is low with respect to work items; and

effectively providing a polling-based mechanism for processing work items, when system utilization is relatively high with respect to work items.

13. (Previously presented) A method as claimed in claim 12 wherein work items are received in accordance with at least one device driver associated with a host system.

14. (Previously presented) A method as claimed in claim 12 wherein the data processing system comprises a storage controller.